

Mesh Learning Using Persistent Homology on the Laplacian Eigenfunctions

Yunhao Zhang ¹, Haowen Liu ¹, Paul Rosen ², and Mustafa Hajij ¹

¹*Ohio State University*
²*University of South Florida*

Abstract

We use persistent homology along with the eigenfunctions of the Laplacian to study similarity amongst triangulated 2-manifolds. Our method relies on studying the lower-star filtration induced by the eigenfunctions of the Laplacian. This gives us a shape descriptor that inherits the rich information encoded in the eigenfunctions of the Laplacian. Moreover, the similarity between these descriptors can be easily computed using tools that are readily available in Topological Data Analysis. We provide experiments to illustrate the effectiveness of the proposed method.

1 Introduction

Shape similarly is a critical problem in computer vision, geometric data processing and computer graphics. Multiple attempts have been made to quantify the similarity among 3D shapes [1, 30, 27]. Several challenges rise up when trying to construct an effective and efficient similarity measure including the complexity of the data, the potential noise in the data and the variation in the structure.

While the Laplacian eigenfunctions have been utilized in the literature of geometric processing to extract shape descriptors [29, 33], most of the eigenfunction-based descriptors require extensive processing to obtain an effective descriptor. Furthermore, the comparison between such descriptors requires designing a specialized similarity measure that adds to overhead computational time [32].

The eigenfunctions of the Laplacian store important information about the geometry of the underlying manifold [23, 24]. Moreover, spaces that have similar structures also tend to have similar sets of eigenfunctions [17]. From this perspective it is natural to utilize the eigenfunctions to measure the similarity between a collection of 3D shapes. The difficulty usually lies in finding the correspondence between two given manifolds [32]. More specifically, when manifold is discretized this correspondence might not even exist due to the difference between the cardinalities of the two vertex sets. Instead sub part correspondence might be considered, which is also a difficult problem [2].

In recent years the interplay between machine learning and Topological Data Analysis (TDA) has witnessed many developments with the better understanding of two tools in TDA: *Persistent Homology* (PH) [8] and the construction of *Mapper* [26]. These TDA tools have been shown to be a powerful tool for shape classification and recognition [15, 21], data summary [12, 5], topological signatures of data [3], graph understanding [13], among others.

In this paper we utilize persistent homology to extract the information encoded in the eigenfunctions of the Laplacian to obtain a topological mesh signature that can be used to measure the similarity among triangulated manifolds. Our proposed method has multiple advantages. On one hand, the method

proposed here avoids the correspondence problem all together. Our approach relies on extracting the topological information encoded into the *lower-star filtration* (see Section 2 for the definition) of the eigenfunctions of the Laplacian and storing the resulting finger print in a structure called the *persistence diagram* [8]. This ultimately allows for an effective comparison between two manifolds by comparing between the persistence diagrams that are induced by the eigenfunctions of the Laplacian.

Using the persistence diagram to compare between metric spaces has been previously applied to meshes [4]. However, the metric-based method in [4] has two major limitations. Firstly, finding the distance function on large meshes is computationally expensive and usually requires utilizing a sampling technique, which might affect the quality of the final persistence diagram. Secondly, in order to obtain a strong descriptor from the persistence diagram induced by the distance matrix, one usually needs the information encoded in higher order persistence diagrams, which are expensive to compute.

Our method avoids these two limitations. On one hand, our method computes the persistence diagram using the lower-star filtration of one or a few eigenfunctions of the Laplacian. In fact we show that utilizing a single eigenfunction yields a persistence diagram that has more classification power than the metric-based approach in [4]. On the other hand, our method only requires the 0-order persistence diagram, which is very efficient to compute. We demonstrate our results by showing the effectiveness of our descriptor on standard datasets. See Section 3 for more details.

2 Persistence Homology on Triangulated Meshes

The mesh topological signature that we propose here utilizes a particular filtration that is induced by a scalar function defined on a mesh M . Our work is mainly aimed at studying triangulated meshes. However, we will state our definitions in terms of simplicial complexes. The reason for this is that most techniques introduced in this article are applicable beyond meshes, and we will provide more details in this regard towards the conclusion.

Let K be a simplicial complex. Let S be an ordered sequence $\sigma_1, \dots, \sigma_n$ of all simplices in the complex K , such that for simplex $\sigma \in K$ every face of σ appears before σ in S . Then S induces a nested sequence of subcomplexes called a *filtration*:

$$\phi = K_0 \subset K_1 \subset \dots \subset K_n = K \quad (2.1)$$

such that $K_i = \cup_{j \leq i} \sigma_j$ is the subcomplex obtained from first i simplices $\sigma_1, \dots, \sigma_i$ of S . Given a filtration as in 2.1, one may apply the homology functor on it to obtain a sequence of homology groups connected by homomorphism maps induced by the inclusions:

$$F(K) : H_d(K_0) \longrightarrow H_d(K_1) \longrightarrow \dots \longrightarrow H_d(K_n) \quad (2.2)$$

A d -homology class $\alpha \in H_d(K_i)$ is said to be *born* at time i if it appears for the first time as a homology class in $H_d(K_i)$. A d -class α *dies* at time j if it is trivial in $H_d(K_j)$ but not trivial in $H_d(K_{j-1})$. The *persistence* of the class α that is born at $H_d(K_i)$ and dies at $H_d(K_j)$ is defined to be $j - i$. *Persistent homology* captures the birth and death events in a given filtration and summarizes them in a multi-set structure called the *persistence diagram* $P^d(K)$ [8]. Specifically, the d -persistence diagram of a filtration $\mathcal{F}(K)$ is a collection of pairs (i, j) in the plane, where each (i, j) indicates a d -homology class that is created at time i in the filtration $\mathcal{F}(K)$ and dies entering time j . A persistence diagram can be represented equivalently by *persistence barcodes* [11]. Specifically every point (i, j) in the persistence diagram can be represented by a *bar* that starts at time u and ends at time v .

Persistence homology tracks the evolution of homology classes as this element moves though the homomorphism from left to right. More specifically, Persistent homology can be defined given any filtration, such as equation 2.1. For our purpose we are given a piece-wise linear function $f : |K| \rightarrow \mathbb{R}$ defined on the vertices of K . We assume that the function f has different values on different vertices of K . Any such a function induces a filtration called the *lower-star* filtration. We define this filtration next.

Let $v \in V(K)$ be a vertex of K . The *star* of v , denoted as $St(v)$, is the set of all simplices in K that contain v as a vertex. When we are given a piece-wise linear function f defined on K , we can also define the lower star of v . Namely, the lower star of a vertex $v \in V(K)$ as $LowSt(v) = \{w \in St(v) | f(w) \leq f(v)\}$.

Let $V = \{v_1, \dots, v_n\}$ be the set of vertices of K sorted in non-decreasing order of their f -values. Let $K_i := \cup_{j \leq i} LowSt(v_j)$. The lower-star filtration is the filtration is defined to be

$$\mathcal{F}_f(K) : \phi = K_0 \subset K_1 \subset \dots \subset K_n = K \quad (2.3)$$

The lower-star filtration reflects the topology of the function f in the sense that the persistence homology induced by the filtration, equation 2.3, is identical to the persistent homology of the sublevel sets of the function f . We denote $P_f^d(K)$ to be the d -persistence diagram induced by the lower-star filtration $\mathcal{F}_f(K)$. In our work, the lower-star filtration is the main tool to extract the signature from a given space.

Here we focus on triangulated meshes, and we only compute the 0-persistence diagram on those meshes using the filtration induced by the lower-star filtration of the eigenfunctions of the Laplacian of these meshes. Such persistence diagrams can be efficiently computed using the *union-find* data structure.

2.1 The Lower-Star Filtration Induced by the Eigenfunctions of the Laplacian.

Let M be an triangulated manifold. The matrix L is self-adjoint and positive semi-definite. It has an *orthonormal eigen-system* $(\lambda_n, \phi_n)_{n=0}^{+\infty}$, $L\phi_n = \lambda_n\phi_n$, with $0 = \lambda_0 \leq \lambda_1 \leq \lambda_{n+1}$, in $C(G)$. The eigenvectors of the Laplacian L form a rich family of scalar functions defined on G that have been utilized extensively in shape understanding and shape comparison [16, 23, 18]. The eigenfunctions of the Laplacian has also been used in graph understand [25], segmentation [22], and spectral clustering [20].

The reasons for extracting the information of the eigenfunctions of the Laplacian using the lower-star filtration can be summarized in the following points:

- The eigenfunctions of the Laplacian provide canonical scalar functions that depend only on the intrinsic geomet-

ric properties of the mesh. In other words, they have all desirable properties of eigenfunctions of the Laplacian—being invariant under certain deformation and robustness to noise and structure variation—will be inherited by the persistence diagram induced by the lower-star filtration of these functions.

- The eigenfunctions of the Laplacian store rich information about the geometry of the underlying manifold and the lower-star filtration provide the means to extract this information and stores it in the structure of the persistence diagram. This structure provides a ranking for features extracted from the eigenfunctions via the notion of persistence.

Ordering the eigenvectors of L by the increasing value of their corresponding eigenvalues, we use the first k -eigenvectors that correspond to the smallest nonzero k eigenvalues of L . These vectors contain low frequency information about the underlying manifold, and they usually retain the shape of complex meshes. In particular, we found that the first non-trivial eigenfunction of the Laplacian to be very effective for our purpose. This eigenfunction, called the *Fiedler vector* [9, 10], has many applications in graph theory as well as in computer graphics [14, 19]. Moreover, this vector has multiple interesting features. For instance, the maximum and the minimum of the Fiedler vector tend to occur at points in the dataset with maximum geodesic distance [6] allowing its values to spread from one end of the graph following its “shape” to the other end.

2.2 Comparing Between Two Persistence Diagrams

We can quantify the structural differences persistence diagrams by using the bottleneck distance.

Let η be a bijection between two persistence diagrams X and Y . The *bottleneck distance* between X and Y [7] is defined as

$$W_\infty(X, Y) = \inf_{\eta: X \rightarrow Y} \sup_{x \in X} \|x - \eta(x)\|_\infty. \quad (2.4)$$

The bottleneck distance requires the persistence diagrams to have the same cardinalities. For this reason we allow infinitely replication of points along the diagonal $y = x$ to a given persistence diagram.

Other distances can also be defined on the space of persistence diagrams, such as the Wasserstein distance. For the purpose of this article we only restrict ourselves with the bottleneck distance.

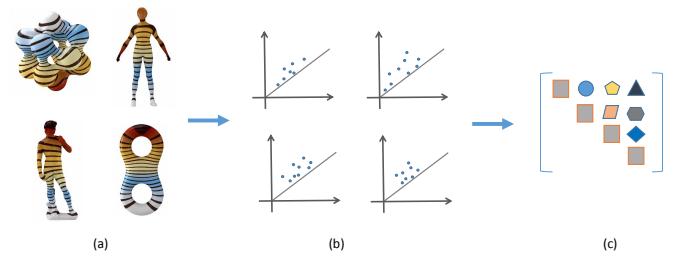


Figure 1: An illustration of the pipeline. (a) We compute one of the eigenfunctions of the Laplacian on the meshes that we want to compare. (b) The lower-star filtrations of the meshes with respect to these scalar functions are computed and their persistence diagrams are extracted. (c) A pairwise comparison between the persistence diagrams is performed using the bottleneck or Wasserstein distances.

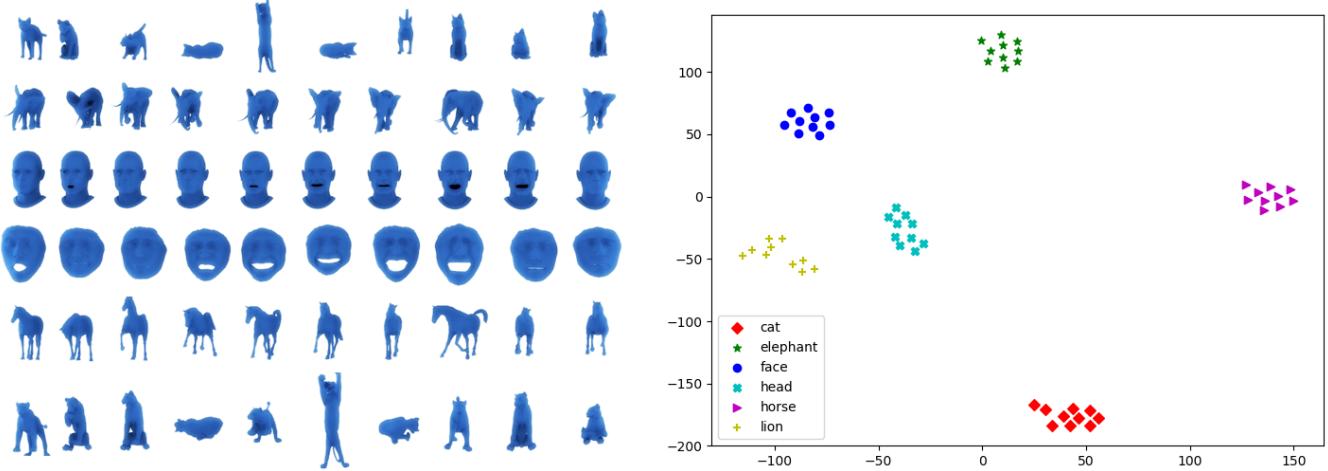


Figure 2: On the left we show the data set that we used in our experiments. The data set consists of 60 triangulated meshes that are divided into 6 categories, which are shown in the figure on the right. We compute the Fielder’s vector for each mesh in this data set and then compute the 0-persistence diagram associated with the lower-star filtration of this vector. The bottleneck distance between these diagrams is calculated, and the figure on the right shows the 2d t-SNE projection obtained using the final distance matrix. Notice how our method provides distinct clusters on this data.

3 Method and Results

Given the above setup, our method can be summarized as follows. First we compute a certain eigenfunction of the Laplacian on a given mesh dataset. In our case we used the Fielder’s vector. We then compute the 0-persistence diagrams of the lower-star filtration induced by the chosen eigenfunction. Once we have the persistence diagrams of the meshes, the distances between these diagrams can be computed using the bottleneck distance we defined in Section 2.2. See Figure 1 for a summary of the method.

To validate the effectiveness of the topological descriptor proposed here, we test it using a publicly available data from [28]. The data set consists of 60 meshes that are divided into 6 categories: cat, elephant, face, head, horse and lion. Each category contains exactly ten triangulated meshes.

On this dataset, we computing the distance matrix between the persistence diagram of the lower-star filtration of the induced Fielder’s vectors using bottleneck distance. To assess the final results, we compute the 2d t-SNE projection [31] of final distance matrix. The result is reported in Figure 2.

Note how this topological descriptor provides a distinct clusters for the underlying data set. Furthermore, the results shown here shows that the proposed descriptor has a better classification power than the one proposed in [4].

One observation worth mentioning here is that the t-SNE projection in Figure 2 shows that heads and faces clusters appear to be closer to the lions cluster than cats cluster. The reason for this is mostly an artifact of the t-SNE projection. In fact the MDS projection shows that the lions and the cats cluster are indeed closer to each other than heads and faces. We presented the t-SNE projection here over MDS since the latter showed the some clusters too close to each other.

4 Further Directions and Conclusion

The experimentation results are only shown with respect to Fielder’s vector. In theory any eigenfunction of the Laplacian can be used in a similar manner, as illustrated above. Combining the signatures obtained from multiple eigenfunction poten-

tially provides even a stronger descriptor. We plan to pursue this direction in the extension of this work.

The construction that we introduced here on triangulated meshes can be easily extended to study similarity between other types of objects. Namely any domain where the definition of the Laplacian is applicable, such as points clouds and graphs. We plan to investigate these directions in the future.

Acknowledgements

This work was supported in part by a grant from National Science Foundation (IIS-1513616).

References

- [1] Silvia Biasotti, Andrea Cerri, Alexander M Bronstein, and Michael M Bronstein. Quantifying 3d shape similarity using maps: Recent trends, applications and perspectives. In *Eurographics (State of the Art Reports)*, pages 135–159, 2014.
- [2] Silvia Biasotti, Simone Marini, Michela Spagnuolo, and Bianca Falcidieno. Sub-part correspondence by structural descriptors of 3d shapes. *Computer-Aided Design*, 38(9):1002–1019, 2006.
- [3] Thomas Bonis, Maks Ovsjanikov, Steve Oudot, and Frédéric Chazal. Persistence-based pooling for shape pose recognition. In *International Workshop on Computational Topology in Image Context*, pages 19–29. Springer, 2016.
- [4] Frédéric Chazal, David Cohen-Steiner, Leonidas J Guibas, Facundo Mémoli, and Steve Y Oudot. Gromov-hausdorff stable signatures for shapes using persistence. *Computer Graphics Forum*, 28(5):1393–1403, 2009.
- [5] ANM Imroz Choudhury, Bei Wang, Paul Rosen, and Valerio Pascucci. Topological analysis and visualization of cyclical behavior in memory reference traces. In *IEEE Pacific Visualization*, pages 9–16. IEEE, 2012.
- [6] Moo K Chung, Seongho Seo, Nagesh Adluru, and Houri K Vorperian. Hot spots conjecture and its application to

- modeling tubular structures. In *International Workshop on Machine Learning in Medical Imaging*, pages 225–232, 2011.
- [7] Herbert Edelsbrunner and John Harer. *Computational Topology: An Introduction*. American Mathematical Society, Providence, RI, USA, 2010.
- [8] Herbert Edelsbrunner, David Letscher, and Afra J. Zomorodian. Topological persistence and simplification. *Discrete and Computational Geometry*, 28:511–533, 2002.
- [9] Miroslav Fiedler. Algebraic connectivity of graphs. *Czechoslovak mathematical journal*, 23(2):298–305, 1973.
- [10] Miroslav Fiedler. A property of eigenvectors of nonnegative symmetric matrices and its application to graph theory. *Czechoslovak Mathematical Journal*, 25(4):619–633, 1975.
- [11] Robert Ghrist. Barcodes: The persistent topology of data. *Bulletin of the American Mathematical Society*, 45:61–75, 2008.
- [12] Mustafa Hajij, Nataša Jonoska, Denys Kukushkin, and Masahico Saito. Graph based analysis for gene segment organization in a scrambled genome. *arXiv preprint arXiv:1801.05922*, 2018.
- [13] Mustafa Hajij, Bei Wang, Carlos Scheidegger, and Paul Rosen. Visual detection of structural changes in time-varying graphs using persistent homology. In *2018 IEEE Pacific Visualization Symposium (PacificVis)*, pages 125–134. IEEE, 2018.
- [14] Martin Isenburg and Peter Lindstrom. Streaming meshes. In *Visualization, 2005. VIS 05. IEEE*, pages 231–238. IEEE, 2005.
- [15] Genki Kusano, Yasuaki Hiraoka, and Kenji Fukumizu. Persistence weighted gaussian kernel for topological data analysis. In *International Conference on Machine Learning*, pages 2004–2013, 2016.
- [16] Stephane Lafon and Ann B Lee. Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization. *IEEE transactions on pattern analysis and machine intelligence*, 28(9):1393–1403, 2006.
- [17] Bruno Levy. Laplace-beltrami eigenfunctions towards an algorithm that “understands” geometry. In *IEEE International Conference on Shape Modeling and Applications 2006 (SMI’06)*, pages 13–13. IEEE, 2006.
- [18] Bruno Lévy. Laplace-beltrami eigenfunctions towards an algorithm that understands geometry. In *IEEE International Conference on Shape Modeling and Applications*, page 13, 2006.
- [19] Patrick Mullen, Yiying Tong, Pierre Alliez, and Mathieu Desbrun. Spectral conformal parameterization. *Computer Graphics Forum*, 27(5):1487–1494, 2008.
- [20] Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in neural information processing systems*, pages 849–856, 2002.
- [21] Jan Reininghaus, Stefan Huber, Ulrich Bauer, and Roland Kwitt. A stable multi-scale kernel for topological machine learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4741–4748, 2015.
- [22] Martin Reuter, Silvia Biasotti, Daniela Giorgi, Giuseppe Patané, and Michela Spagnuolo. Discrete laplace–beltrami operators for shape analysis and segmentation. *Computers & Graphics*, 33(3):381–390, 2009.
- [23] Martin Reuter, Franz-Erich Wolter, and Niklas Peinecke. Laplace–beltrami spectra as shape-dna of surfaces and solids. *Computer-Aided Design*, 38(4):342–366, 2006.
- [24] Raif M Rustamov. Laplace–beltrami eigenfunctions for deformation invariant shape representation. In *Proceedings of the fifth Eurographics symposium on Geometry processing*, pages 225–233. Eurographics Association, 2007.
- [25] David I Shuman, Sunil K Narang, Pascal Frossard, Antonio Ortega, and Pierre Vandergheynst. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE Signal Processing Magazine*, 30(3):83–98, 2013.
- [26] Gurjeet Singh, Facundo Mémoli, and Gunnar E Carlsson. Topological methods for the analysis of high dimensional data sets and 3d object recognition. In *SPBG*, pages 91–100, 2007.
- [27] Tomáš Skopal and Benjamin Bustos. On nonmetric similarity search problems in complex domains. *ACM Computing Surveys (CSUR)*, 43(4):34, 2011.
- [28] Robert W Sumner and Jovan Popović. Deformation transfer for triangle meshes. *ACM Transactions on graphics (TOG)*, 23(3):399–405, 2004.
- [29] Jian Sun, Maks Ovsjanikov, and Leonidas Guibas. A concise and provably informative multi-scale signature based on heat diffusion. *Computer graphics forum*, 28(5):1383–1392, 2009.
- [30] Gary KL Tam, Zhi-Quan Cheng, Yu-Kun Lai, Frank C Langbein, Yonghuai Liu, David Marshall, Ralph R Martin, Xian-Fang Sun, and Paul L Rosin. Registration of 3d point clouds and meshes: a survey from rigid to nonrigid. *IEEE transactions on visualization and computer graphics*, 19(7):1199–1217, 2013.
- [31] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.
- [32] Oliver Van Kaick, Hao Zhang, Ghassan Hamarneh, and Daniel Cohen-Or. A survey on shape correspondence. *Computer Graphics Forum*, 30(6):1681–1707, 2011.
- [33] Lisha Zhang, Manuel João da Fonseca, Alfredo Ferreira, and Combinando Realidade Aumentada e Recuperação. Survey on 3d shape descriptors. *Fundação para a Ciência e Tecnologia, Lisboa, Portugal, Tech. Rep. Technical Report, DecorAR (FCT POSC/EIA/59938/2004)*, 3, 2007.